Remarks

Reconsideration of the application as amended herein is respectfully requested. Claims 1-4 are pending in this application. Claim 1 has been amended. No claims have been added or cancelled.

Claim Rejections – 35 U.S.C. §112

Claim 1 was rejected under 35 U.S.C. § 112, second paragraph, as being indefinite for failing to particularly point out and distinctly claim the subject matter which applicant regards as the invention. Specifically, the Examiner asserted that the term "external data" in claim 1 is a relative term which renders the claim indefinite.

Applicants have amended claim 1 to remove "external" from the term "external data." Therefore, Applicants respectfully submit that the rejection under 35 U.S.C. § 112 has been overcome.

Claim Rejections – 35 U.S.C. §103

Claims 1-4 were rejected under 35 U.S.C. § 103(a) as being unpatentable over Beausoleil et al., U.S. Patent No. 5,551,013 (hereinafter "Beausoleil") in view of Austin et al., U.S. Patent No. 4,885,684 (hereinafter "Austin") and further in view of Baker et al., U.S. Patent No. 5,701,502 (hereinafter "Baker"). Applicants respectfully traverse.

Claim 1 has been amended to make clear that "the **module main memory unit** is **separate** from the **control store**." (emphasis added). Support for this amendment can be found, for example, on Figure 2, which clearly shows the module main memory unit 18 being separate from the control store 30. Beausoleil, Austin and Baker, either alone or in combination, do not teach or suggest all of the limitations of Claim 1 as amended.

Beausoleil discloses a emulation processor that emulates both logic functions and memory array operations of a VLSI chip design. *See Beausoleil*, Col. 2, Lines 27-34 and Col. 3,

lines 38-48. During each step of a sequencer, the emulation processor emulates either a logic function or a memory array operation of the design. *See Beausoleil*, Col. 5, Lines 47-53. Beausoleil further discloses a memory operation (MOP) bit that is included in a control word. When the MOP bit is "inactive (off)," the emulation processor emulates a logic function that is defined by a left and right control word pair in the left and right control stores. When the MOP bit is "active (on)," the emulation processor emulates a memory operation that is defined by a left control word in the left control store. During the memory operation, the right control store is addressed by a combination of data from the left control word and a memory address register, and the contents of the right control word are interpreted as data in the emulated memory array. *See Beausoleil*, Col. 6, Lines 14-27. Therefore, the MOP bit controls whether the emulation processor will emulate a logic function or a memory array operation of the VLSI chip design. At most, the MOP bit controls how the emulation processor accesses the control store based on whether the emulation processor is to emulate a logic function or a memory operation. The MOP bit has nothing to do with controlling access to a module main memory unit that is **separate** from the control store as required by amended claim 1. Therefore, the MOP bit of Beausoleil fails to teach or suggest, either explicitly or implicitly, a blocking code in a programmable sequence of emulation steps that blocks access to a module main memory unit that is **separate** from the control store.

Like Beausoleil, Austin and Baker also fail to teach or suggest a blocking code in a programmable sequence of emulation steps that blocks access to a module main memory unit that is **separate** from the control store. Because Beausoleil, Austin and Baker, either alone or in combination, fail to teach or suggest this claim limitation, Applicants submit that Claim 1 is patentable over the cited references.

Further, it would not have been obvious to one skilled in the art to use the MOP bit of Beausoleil to interrupt emulation and initiate maintenance operations in view of Austin's maintenance bus or Baker. To begin, emulation is **not** interrupted by the MOP bit of Beausoleil. In fact, the emulation processor of Beausoleil continues to perform emulation during both states of the MOP bit. When the MOP bit is inactive, the emulation processor emulates a logic function, and when the MOP bit is active, the emulation processor emulates a memory operation.

Therefore, interrupting emulation based on the state of the MOP bit (active or inactive) runs contrary to the explicit teaching of Beausoleil. Further, the MOP bit is completely unrelated to whether maintenance is needed. As explained above, the MOP bit signals whether the emulation processor of Beausoleil will emulate a logic function or a memory operation of the VLSI chip design being emulated. When the MOP bit is inactive, the emulation processor emulates a logic function, and when the MOP bit is active, the emulation processor emulates a memory operation. The state of the MOP bit (active or inactive) is completely unrelated to whether maintenance is needed. As a result, one skilled in the art would not use the MOP bit to initiate maintenance operations (e.g., re-compiling code). In fact, using the MOP bit to initiate maintenance operations would produce highly undesirable results. Beausoleil discloses that the emulation module has the potential to emulate up to a quarter of a million bits of memory array data. *See Beausoleil*, Col. 3, Lines 61-65. Emulating such a large memory array requires tens of thousands of active MOP bits in the emulation program. If a system used the active MOP bits to initiate maintenance operations, then the system would perform tens of thousands of maintenance operations during a single emulation cycle, which would severely slow down the system. In short, the MOP bit of Beausoleil is used during emulation to signal whether the emulation processor will emulate a memory operation or a logic function of the VLSI chip design being emulated, which is completely unrelated to whether the system needs maintenance. Therefore, one skilled in the art would not use the MOP bit of Beausoleil to initiate maintenance operations in view of Austin's maintenance bus or Baker.

Further, one skilled in the art would not combine Beausoleil and Baker, as explained below. Baker discloses fault-tolerant hardware synchronization for synchronizing processing units in a partner pair. One of the processing units, referred to as the "master," is responsible for synchronization and its partner which undergoes synchronization is referred to as the "slave." *See Baker*, Col. 118, Line 47-57. Baker further discloses that the synchronization requirement is "invoked by an Initial Power On, the appearance of a new partner or a recovery from an error condition that caused two existing partners to lose synchronization (each case forcing a Maintenance Interrupt)." *See Baker*, Col. 119, Lines 4-10. Therefore, the maintenance interrupt, which signals that synchronization is required, occurs at initial power on, when a new partner appears or when an error condition causes two existing partners to lose synchronization.

One skilled in the art would not use the MOP bit of Beausoleil to signal the maintenance interrupt of Baker, and hence the requirement for synchronization. This is because the MOP bit of Beausoleil is completely unrelated to whether synchronization is required. As explained above, the MOP bit signals whether the emulation processor of Beausoleil will emulate a logic function or a memory operation of the VLSI chip design. When the MOP bit is inactive, the emulation processor emulates a logic function, and when the MOP bit is active, the emulation processor emulates a memory operation. The state of the MOP bit (active or inactive) is completely unrelated to the conditions requiring synchronization as taught by Baker, which are initial power on, appearance of a new partner or an error condition that causes two existing partners to lose synchronization. Therefore, the MOP bit of Beausoleil is completely unsuitable for signaling the maintenance interrupt. In fact, doing so would result in a non-functional system because a system relying on the MOP bit of Beausoleil to signal the maintenance interrupt would have no way of knowing when synchronization is actually required. Such a combined system would perform synchronization based on whether the MOP bit signals a logic function emulation or a memory operation emulation and **not** on whether synchronization is actually required. As a result, there would be instances where the combined system would fail to perform synchronization when required. Because the MOP bit of Beausoleil is completely unrelated to whether synchronization is required as taught by Baker, one skilled in the art would not combine Beausoleil and Baker.

For the above reasons, Applicants submit that Claim 1 is patentable over the cited references and respectfully request that the rejection of Claim 1 be withdrawn.

Dependent Claims 2-4

The foregoing arguments apply to Claims 2-4, as they all are dependent on Claim 1. Therefore, Applicants respectfully submit that Claims 2-4 are allowable as well.

Conclusion

Applicants respectfully submit that this application is in condition for allowance, which is respectfully requested. Should the Examiner have any questions or comments on the application, the Examiner should feel free to contact the undersigned via telephone.

The Commissioner is authorized to charge any fee which may be required in connection with this Amendment to deposit account No. 15-0665.

Respectfully submitted,
ORRICK, HERRINGTON & SUTCLIFFE LLP

Dated: 02/14/06

By: Eugene Worley
Eugene Worley
Reg. No. 47,186

Orrick, Herrington & Sutcliffe LLP
4 Park Plaza, Suite 1600
Irvine, CA 92614-2558
Tel. 949-567-6700
Fax: 949-567-6710